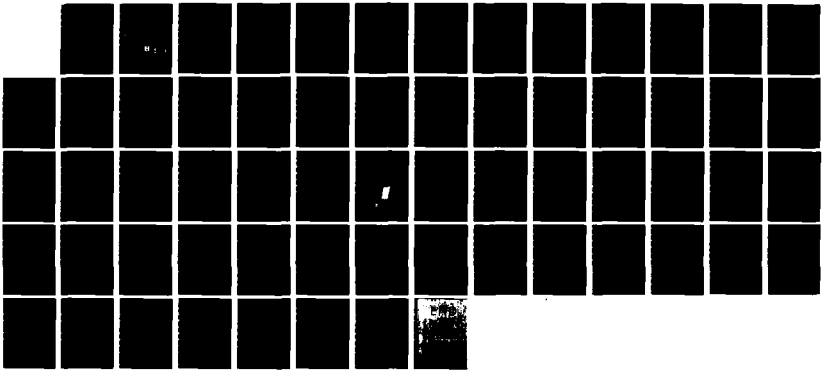
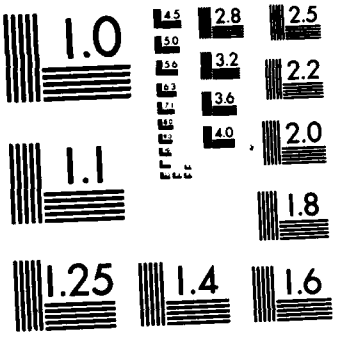
AD-A134 930 HQ AFSC SELECTION OF A MICROPROCESSOR DEVELOPMENT SYSTEM(U) MITRE CORP BEDFORD MA C H PRICE OCT 83 MTR-8866 ESD-TR-83-205 F19628-82-C-0001 1/1

UNCLASSIFIED F/G 5/1 NL





(12)

ESD-TR-83-205

MTR-8866

AD-A134930

HQ AFSC SELECTION OF A MICROPROCESSOR DEVELOPMENT SYSTEM

By
C. H. PRICE, JR.

OCTOBER 1983

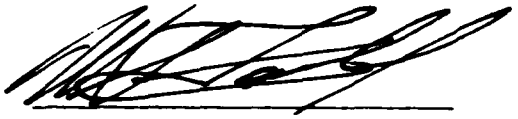
Prepared for
DEPUTY FOR MISSION SUPPORT SYSTEMS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
Hanscom Air Force Base, Massachusetts

DTIC FILE COPY

Project No. 479C
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract No. F19628-82-C-0001

REVIEW AND APPROVAL

This technical report has been reviewed and is approved for publication.



W. T. FARRELL, Lt Col, USAF
Deputy Director, Automated Management Systems
Deputy for Mission Support Systems

FOR THE COMMANDER

ROBERT J. KENT, GM-15
Director, Automated Management Systems
Deputy for Mission Support Systems

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: ESD-TR-83-205	2. GOVT ACCESSION NO.: AD A134930
3. RECIPIENT'S CATALOG NUMBER:	
4. TITLE *(and Subtitle)*: HQ AFSC SELECTION OF A MICROPROCESSOR DEVELOPMENT SYSTEM	5. TYPE OF REPORT & PERIOD COVERED:
6. PERFORMING ORG. REPORT NUMBER: MTR-8866	
7. AUTHOR(s): C. H. PRICE JR.	8. CONTRACT OR GRANT NUMBER(s): F19628-82-C-0001
9. PERFORMING ORGANIZATION NAME AND ADDRESS: The MITRE Corporation, Burlington Road, Bedford, MA 01730	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Project No. 479C
11. CONTROLLING OFFICE NAME AND ADDRESS: Deputy for Mission Support Systems, Electronic Systems Division, AFSC, Hanscom Air Force Base, MA 01731	12. REPORT DATE: OCTOBER 1983
	13. NUMBER OF PAGES: 62
14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*:	15. SECURITY CLASS. *(of this report)*: UNCLASSIFIED
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

DMS COPY INSPECTED

Accession For
NTIS GRA&I
DTIC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

LOCAL AREA NETWORK
MICROPROCESSOR-BASED DESIGNS
MICROPROCESSOR DEVELOPMENT SYSTEM
SYSTEM DESIGN LABORATORY
VENDOR SELECTION CRITERIA

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

A microprocessor development system is essential to the system design laboratory where it is used to develop, test, and debug microprocessor-based circuitry and software. This report describes the functions and components of a typical development system, provides vendor selection criteria, and compares the characteristics of three proven or representative systems to determine the one most suitable for acquisition by HQ AFSC.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

ACKNOWLEDGEMENTS

The author gratefully acknowledges the technical support and assistance of the people who helped bring this document to publication: Virginia F. Haydu for her highly respected professional editing, Joan R. Irwin for her assiduous typing to bring order out of chaos, and to Michele B. Sullivan for her accurate preparation of graphics.

This report was prepared by the MITRE Corporation under Project No. 479C. The contract is sponsored by the Electronic Systems Division, Air Force Systems Command, Hanscom Air Force Base, Massachusetts.

TABLE OF CONTENTS

TABLE OF CONTENTS (Concluded)

LIST OF FIGURES

INTRODUCTION

SECTION I

1.1 BACKGROUND

This report is part of a study made to fulfill the requirements for providing a local area network design for Headquarters, Air Force Systems Command, Andrews Air Force Base. The local area network, LNET, is intended to provide communications within a limited geographic area between dissimilar terminals and computers. The interface between these system devices and the network transmission medium is a Bus Interface Unit (BIU). The BIU is a microprocessor-based device which can handle most of the strictly defined procedures (protocols) that govern host-terminal communication. However, to enable the previously installed Burroughs and Honeywell equipment to communicate via the network, their special protocols must be accommodated. These and other special protocols require the development of additional software and hardware which must be integrated within the BIU design. Additional microprocessor-based circuitry to be designed by HQ AFSC will allow commercial BIUs to handle these special protocols.

The primary tool for implementing microprocessor-based designs is a microprocessor development system. This is a system that enables the designer to develop, test, and debug the software and circuitry of microprocessor-based prototype systems. In this way, a microprocessor development system can save considerable time, effort, and expense.

1.2 SCOPE

This report provides the rationale and criteria for selecting a microprocessor development system from a number of development systems readily available in the marketplace. The marketing strategies of competing companies producing development systems are examined, and the characteristics of three proven systems are compared to determine which of these systems should be acquired by HQ AFSC. Each of the systems has all of the fundamental attributes necessary for microprocessor system designers to successfully implement their designs.

After this introduction, section 2 begins with a brief history of the evolution of the need for microprocessor development systems in industry and describes the development system in terms of its functions, components, and importance to the design cycle. Section 3 contrasts single-vendor, dedicated systems with universal systems. Section 4 provides a set of selection criteria and compares the GenRad, Hewlett-Packard, and Tektronix systems. Section 5 contains conclusions and recommendations. System comparison charts and selected vendor addresses can be found in appendices. A glossary of relevant terms may be found at the end of the document (reference 1).

SECTION 2

MICROPROCESSOR DEVELOPMENT SYSTEM DESCRIPTION

2.1 DEVELOPMENT SYSTEMS IN INDUSTRY

The microprocessor development system came into being in the early seventies because system designers and microprocessor chip vendors needed more sophisticated tools and electronic instrumentation to help them implement new microprocessor applications and bring their products to the marketplace quickly and relatively bug free. The microprocessor and peripheral support chips required by new prototype systems proved extremely difficult to evaluate and debug using standard checkout procedures and equipment such as oscilloscopes, logic analyzers and assorted test devices. The microprocessor development system now gives designers new design, test, and debug capabilities which were previously unavailable (reference 2). Today's advantages gained from using a development system are a direct result of the relentless competitive pressure on the microcircuit industry to design better means of using their products (reference 3). Accordingly, new development systems and system enhancements are appearing on the marketplace at a rapid rate.

Development system manufacturers are constantly being requested to improve their software packages by making them more powerful and easier to use. These requests are brought on by the spiraling cost of software design for new prototype products. Typical development cost studies for any product design over the period 1979-1982 show hardware costs decreasing by approximately 20% per year. Over the

same period, software costs rose at approximately 12% per year (reference 4). Thus, it makes sense for development system vendors to produce an efficient and "friendly" software package to go along with a good hardware package (reference 5).

2.2 MICROPROCESSOR DEVELOPMENT SYSTEM FUNCTIONS

The capabilities summarized below can be provided by a typical development system and are what make microprocessor development systems essential for development, test and debugging of microprocessor-based circuitry and systems.

2.2.1 Software Development

Operating system. Each development system has a disk-based operating system which provides the user with edit, assemble/compile, link, locate, and debug facilities (reference 6). Recently designed operating system software is "user friendly" and includes "HELP" files and menu-driven utility files to aid the programmer. Three vendors, Gould/Millenium, Tektronix and Zilog, have based their new operating systems on a version of UNIX in an effort to produce an easy to use, powerful operating system. These UNIX-based systems acquire some very useful features:

- Multi-tasking...allows users to run several tasks simultaneously.

- Hierarchical file structure...new files can be easily added to an existing file structure (reference 7).

New enhancements to UNIX include HELP features to improve system/user interaction, including prompts and on-screen aids (reference 8).

Editor. The user creates and modifies source programs by employing editor commands to change, delete, insert, read, write, move, and display code and data. The most efficient text editors provide two distinct modes of operation: test mode and command mode. Text mode makes full use of the CRT display with its movable cursor, insertion and deletion modes. Command mode implements text editing primitives such as string search, block change or replacement and formatting (reference 9).

Assembler. An assembler is a utility which converts mnemonic source code written by the user into absolute object code which is readable and executable by the microprocessor. Some units include a relocatable macro assembler which allows the programmer to define a software operator, resulting in a number of assembly language statements to be executed.

Compiler. The system compiler accepts a source program written in a high level language and converts it into object code. Optional high-level compilers with BASIC, FORTRAN, PASCAL, or PL/1 are available. Some of the system manufacturers have developed their own versions of PL/1. These special languages have been supported solely by that manufacturer, e.g., PL/M (Intel), PL/Z (Zilog), and MPL (Motorola).

Linking Loader. The linker is a program that reads one or more separate relocatable object files called for by the user and binds their procedures and data into one executable image file. All addresses in the relocatable files are recalculated and assigned relative to a given base address. This base address is known only when the program is actually loaded into main memory. The loader is

responsible for recomputing this address at load time and for linking program calls to actual locations of subprograms in memory (references 10, 11).

Command File. This capability stores a series of operation keyboard commands which can be subsequently invoked by a single command.

2.2.2 System Testing

In-Circuit Emulation. The emulation system will allow the designer to run system software with no existing prototype hardware. The emulation system memory within the development system can be allocated to substitute for prototype memories, i.e., if portions or all of prototype memory are missing at the time of test, memory segments from the development system can be used. Very high speed logic examines each memory reference command and switches memory access to either prototype memory or development system memory. This activity occurs within the regular cycle time of the microprocessor, hence the operation remains transparent to overall system performance.

Logic Analyzer. The integrated logic analyzer transparently monitors the emulator bus for a real-time view of activity on address, data, status and control lines of the target (prototype) microprocessor (reference 12).

Monitor Step Mode/Trace Mode. Single step mode allows the prototype microprocessor to single step through each command. Trace mode keeps a record of every program step.

2.2.3 Test and Debug

From the control console, the user is able to inspect and control each step of the program as it will execute in the prototype system:

- o Load the prototype memory with the developed object code.
- o Control the starting and stopping operation of the microprocessor.
- o Single step the microprocessor through the developed program.
- o Examine the system bus conditions during prototype circuit operations.
- o Examine the microprocessor internal registers during prototype circuit operations. This mode of operation is used primarily for correcting software problems. The microprocessor has to dump the contents of its internal registers during this operation. The prototype system is forced to wait while the microprocessor reports on its internal register contents as each step of the program is executed. This waiting sequence is a common occurrence with every development system and CPU (reference 13).

Interactive Debugger. The interactive debugger is a program that can help the user analyze the interaction between the software and hardware. This analysis includes information with respect to address, data and control lines, software interrupts, I/O events, and selected nodes. The program can accept commands such as store,

execute, find data, set breakpoint, etc., and can generate full screen hexadecimal memory dumps. Most development system software also includes symbolic debugging capability.

2.2.4 Additional Microprocessor Development System Capabilities

- o Real time emulation for all supported microprocessors with all memory configurations. The development system is transparent to the prototype systems under test.

- o 16-Bit microprocessor upgrades easily achieved.

- o Full screen editor - cursor control editing provides time-saving features over a line editor.

- o Split screen display - shows current program status as well as program text.

- o Memory allocation - allows designation of any area of system memory to be used as prototype memory.

- o Memory map/write protect - can designate selected areas of memory as read-only memory.

- o Log capability - Can store keyboard commands for future reference in the same sequence as they were issued.

- o RS-232C interface - provides connection of any RS-232C terminal to system. This feature is especially useful for hard copy processes.

- o Diagnostics - reduce system downtime by providing diagnostic information about what has malfunctioned.

- o Help files - provide immediate, specific on-line instruction and thus reduce dependency on system manuals.

- o Software service calls - give the user program control of I/O devices and enable the user to accomplish I/O read/writes.

2.3 DEVELOPMENT SYSTEM HARDWARE

A microprocessor development system generally comprises the following hardware:

a. Central Processing Unit (CPU) executes the various algorithms required for the development tasks, including editcr, assembler, debugger.

b. System Memory - usually transparent to the user; contains host programs such as the editor, assembler, compiler, debugger and system utility workspace.

c. User Memory - available from 16K to 64K bytes providing a corresponding amount of user-addressable space.

d. Control Console - provides an interface between the system designer and the development system. Operator commands are entered on the keyboard, and a display provides feedback from the system utilities and the prototype circuits.

e. Dual double-density floppy disk, hard disk drive, or a combination of both provide mass storage for operating programs, designer programs, temporary or permanent data and system diagnostics.

f. In-Circuit Emulator - consists of the interchangeable emulator card internal to the development system and an external electronics pod containing the microprocessor to be emulated. The two are connected by an umbilical cable. The emulator card and pod must be changed to achieve emulation of different microprocessors.

g. Logic Analyzer - The logic analyzer monitors the flow of electronic signals in the prototype circuitry. It is built-in and software controllable.

h. PROM Programmer - provides programming capability to write and read most PROMs.

2.4 MICROPROCESSOR-BASED PROTOTYPE SYSTEM DESIGN STEPS

The design of a prototype microprocessor-based system typically proceeds through the steps outlined below and illustrated in figure 1 (reference 14). The contribution of a microprocessor development system to the timely implementation of the prototype system design grows more and more important as the design nears completion.

a. Project Definition. When the project is defined, hardware and software development tasks are assigned and must then proceed in parallel to make the most efficient use of time and equipment.

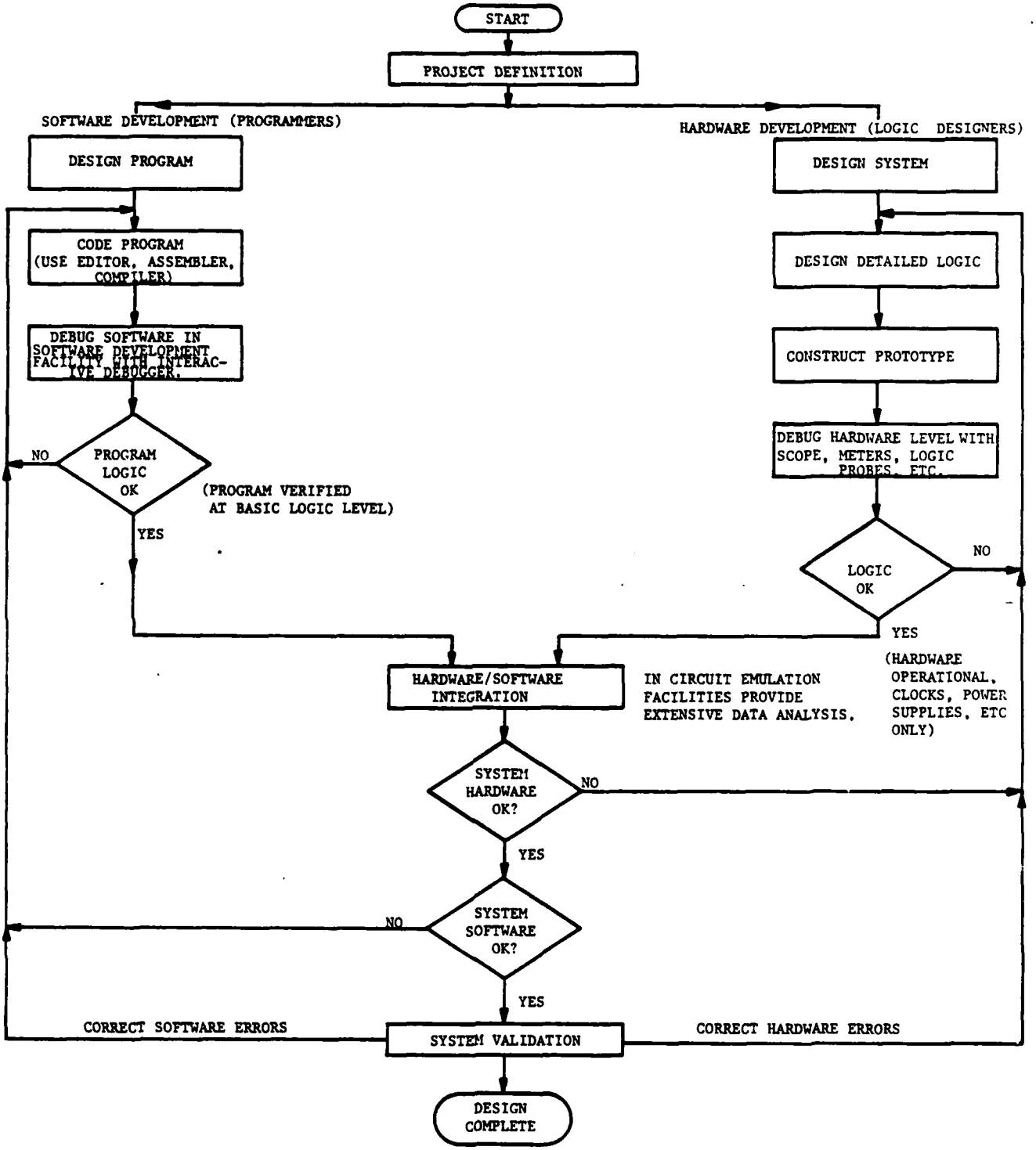


Figure 1. PROTOTYPE SYSTEM DESIGN SEQUENCE

b. Software Development. Program designers and diagnostic engineers layout the software architecture and allocated programming tasks into the development of small program modules which can be efficiently tested and linked together at run time.

c. Coding. The programs that will control the microprocessr and prototype circuitry are written using the development system editor and assembler/compiler, or using similar facilities on an in-house mainframe computer.

d. Debug Software. The object code resulting from the programming effort is loaded into the development system memory or the prototype memory or any segment of either memory for software testing and debugging.

e. Hardware Development. At the same time the software is being written, the logic designers are proceeding from the conceptual black box modules which encompass entire design tasks to increasingly smaller design modules which perform specific tasks.

f. Detailed Logic Design. The exact operation of each module is detailed to produce a functional diagram of the circuitry which includes voltage and signal levels from specified integrated circuits and discrete components.

g. Prototype Construction. The prototype construction proceeds according to detailed drawings created during the logic design phase.

h. Debug Hardware. Each prototype module is tested and debugged to ensure that it operates according to the detailed plans of the logic designer.

i. Software/Hardware Integration. This is the most difficult phase of the project. The complexity of the software and hardware functional interaction may easily produce multiple results, each of which tends to mask the true cause of the problem. All of the microprocessor development system capabilities will be utilized to run the system, monitor performance, and debug faults (reference 15).

j. Iterative Test and Checkout. The microprocessor development system is the primary development tool during this space of system checkout. An undefined number of iterations will be necessary before the system design is complete.

SECTION 3

DEVELOPMENT SYSTEM MARKETING

3.1 SINGLE VENDOR DEDICATED SYSTEMS

When a large microcircuit corporation manufactures a microprocessor development system, the system is dedicated to the family of chips manufactured by that corporation. Intel, for example, offers a versatile line of microprocessors and development system tools. However, the Intel development system line will not handle Motorola, Rockwell, Texas Instruments, or Zilog microprocessors.

Microcircuit vendors are continually bringing forth newer, faster, more powerful chips having specifications system designers may examine with enthusiasm. This presents a dilemma for the system design team that has purchased a single-vendor dedicated system. Afer designers buy a development system, they invest considerable time learning to use the equipment. Even though constant technology push may require them to think about switching microproduct vendors, they are reluctant to invest additional time learning the vagaries of yet another development system. Therefore, even though another vendor may have come out with newer, more powerful microcircuits that the designers would like to incorporate into future systems, the designers continue to select new microprocessors from the family of chips compatible with their existing dedicated development system.

After proceeding through several designs based on the family of chips and the same development system vendor, the design team becomes increasingly locked into that vendor and the vendor's family

of microproducts. This sequence has occurred many times (references 4, 16). Thus, the microcircuit vendor marketing strategy has been successful, i.e., dedicated and committed users have been gained, but this success is not necessarily advantageous for the user.

3.2 UNIVERSAL DEVELOPMENT SYSTEM

The difficult question then becomes, "How can we acquire a development system which can accurately support microproducts from several different manufacturers?" The need for a universal development system would seem to rule out purchasing a system geared to the microproducts of one vendor. An alternate choice for the system designers would be to purchase a development system from a measurement/instrument house.

Supplying a development system which will provide a very high level of design support for any number of microprocessors is a difficult engineering task, but instrument/measurement vendors have been able to provide well-engineered development systems that are compatible with many types of microprocessors. These universal systems allow the designers more flexibility, because they can choose a microprocessor suitable for the design rather than create a design suitable for use with a given microprocessor (references 4, 17).

This universality is not attained without risk, however. The instrument/measurement house designers must wait until the new chip is available before they can produce electronics that will emulate it. This means our designer also has to wait. In many cases, this delay can be tolerated because the chip manufacturer typically will take that long to provide full documentation for the new device.

When the various competing development system manufacturers are contacted regarding their specifications for equipment functionality, universality and costs, universality tends to be the parameter that separates the microprocessor vendors from the instrument vendors. System universality, that is, the capability of using the development system for a varied selection of microprocessors from different families, is not emphasized by the microprocessor vendors. This same feature is emphasized by the instrument/measurement vendors for obvious reasons (reference 14).

3.3 COMPARISON OF SYSTEM ARCHITECTURES

The basic architecture of the universal development system is essentially the same as that of the single vendor system, i.e., they both employ a multiple bus system (figure 2). To change system support from one microprocessor to another, both kinds of systems require changing the emulation card in the chassis and the emulation interface electronic package which is housed in a pod external to the development system. The pod contains a carefully selected microprocessor of the type to be used in the prototype electronic package under development. When the prototype microprocessor is removed from its circuit and is replaced by the emulator microprocessor and electronics, ideally total transparency should occur, resulting in identical prototype operational behavior (reference 15).

The universal development system ideally will achieve this transparency for a wide range of microprocessors from multiple vendors. The single vendor development system will produce similar results for a number of microproducts strictly from the same vendor.

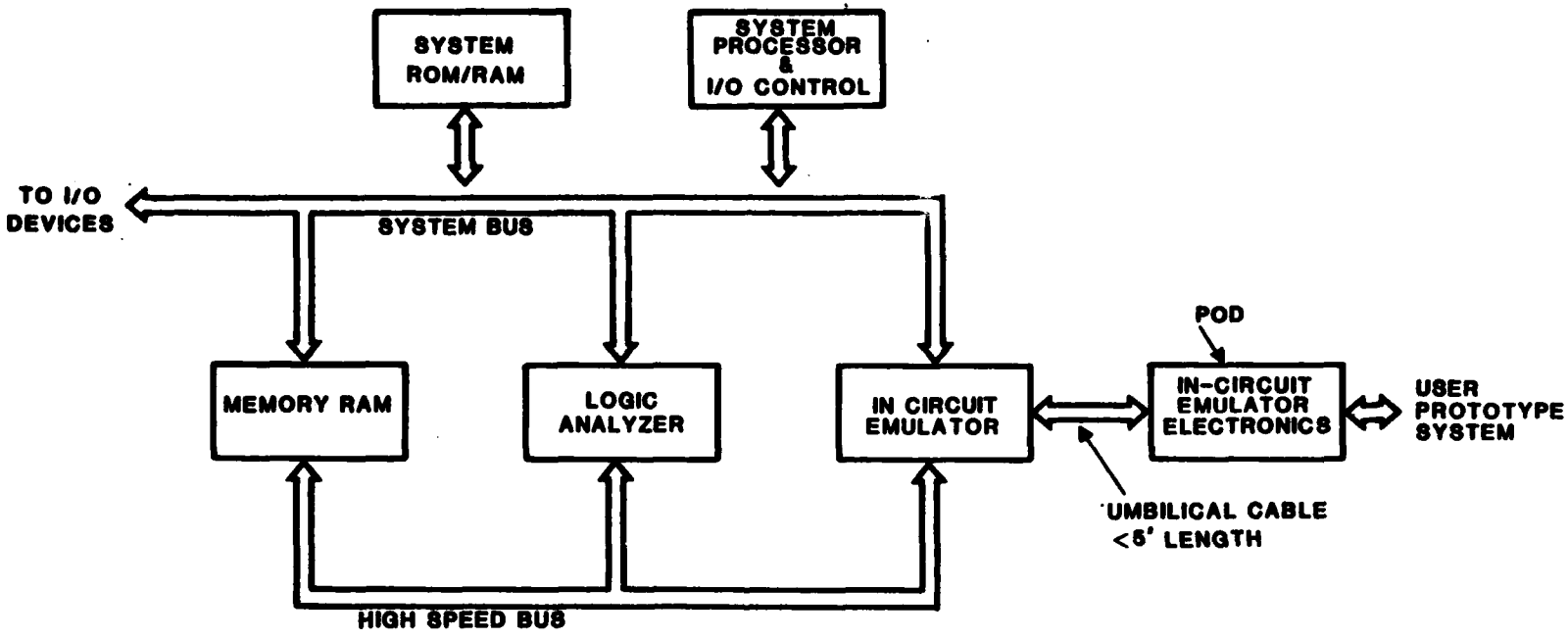


Figure 2. Multiple Bus Development System Architecture

3.4 TYPICAL MULTIPLE BUS DEVELOPMENT SYSTEM ARCHITECTURE

The multiple bus development system design based on multiple microprocessors driving a multiple bus architecture, provides nearly perfect high-speed emulation. The system bus is controlled by the host system processor which handles all system utilities. The high speed bus, on which the prototype microprocessor runs, provides emulation transparent to most, if not all, prototype circuit functions.

Universal support for a wide range of microprocessors is accomplished by allowing each emulated microprocessor to access high-speed memory via a high-speed (10-15 MHz) memory bus. This technique allows the target microprocessor to operate independently of the system processor. Bus contention problems are thus eliminated. Multiple workstations, each having a separate high-speed bus, memory, and emulator components can be supported.

A multiple workstation arrangement is typically driven by a host station supported by a hard disk (reference 18). Each workstation can support development work on a different microprocessor. The universal development systems can support development work on different microprocessors from different vendors at each station simultaneously.

Emulation is accomplished by using an exact prototype of the microprocessor under development. This unit is located in a pod external to the system electronics. An emulator card containing electronic circuitry specifically for the prototype microprocessor

is housed in the development system chassis. The emulator card, pod, and the prototype circuit are connected by high-speed umbilical cables. A short (less than five feet) umbilical cable connects the microprocessor in the pod to the emulator card. A similar cable from the pod connects the microprocessor directly to the prototype circuit bus.

The umbilical cable from the pod is terminated with a plug which replaces the prototype microprocessor in the system under test. The microprocessor emulator provides the designer with a powerful tool for controlling interaction with the prototype hardware and software from the development system keyboard.

SECTION 4

MICROPROCESSOR DEVELOPMENT SYSTEM COMPARISON

In this section, selected hardware, software and emulation features of three major representative microprocessor development systems will be examined. In some cases, special enhancements introduced by other manufacturers will be noted as desirable options to be acquired when selecting a development system. For instance, the PL/M-86 compiler produced by Intel gives the user a menu for selecting specific compilation techniques to optimize the following:

- o Amount of code generated
- o Execution speed of the code
- o Both amount of code generated and execution speed.

Particular functions such as the one described probably will not be available from every vendor, but they will provide a good basis for comparison.

4.1 MICROPROCESSOR DEVELOPMENT SYSTEMS CRITERIA

The three representative development systems analyzed in this section have many of the same properties and meet the following criteria. They are produced by electronics measurements firms with relatively long histories of producing high-quality electronic equipment. Each manufacturer has delivered at least two generations of development systems nation-wide. Each system provides complete hardware and software capability for implementing a microprocessor-

based design. Each is an integrated development system produced and supported by a single manufacturer (see appendices I and II) and each supports microprocessors from many different vendors.

GenRad.....2300 Series Advanced Development Systems
Hewlett-Packard...64000 Series Logic Development System
Tektronix.....8500 Series Microcomputer Development Lab

4.2 GENRAD

The GenRad systems discussed in this section are an advanced multibus, multiprocessor series, the 2300, and an entry level single processor system, the Low-Cost Development System. The system manufactured by FutureData/Kontron Electronics, formerly owned by GenRad, is also discussed in this section.

4.2.1 Advanced Development System (SECU)

The GenRad Advanced Development System features a control console, floppy disk, drives, and a Slave Emulator Control Unit (SECU). The SECU is a multiprocessor, multibus emulation system which provides independent and simultaneous operation of a prototype system processor. From the console, the user can control simultaneous microprocessor emulation of up to eight different microprocessors running at real time speeds (each SECU controls one microprocessor). An RS-232C full duplex interface bus (19.2K bits/sec.) ties each SECU to the system. Each SECU contains a Z80 interface processor which handles debugging requests, emulation controls, memory mapping, etc., and handles development system

requests via a high level protocol language. A high-speed RAM (128K, 150 ns access time) is available in each SECU. The SECU permits nonstop, full speed transparent emulation for either 8 or 16-bit microprocessors:

- o Intel 8080A, 8085A, 8086
- o Zilog Z80, Z8001, Z8002
- o Motorola 6800, 6802, 6809, 6809E, 68000

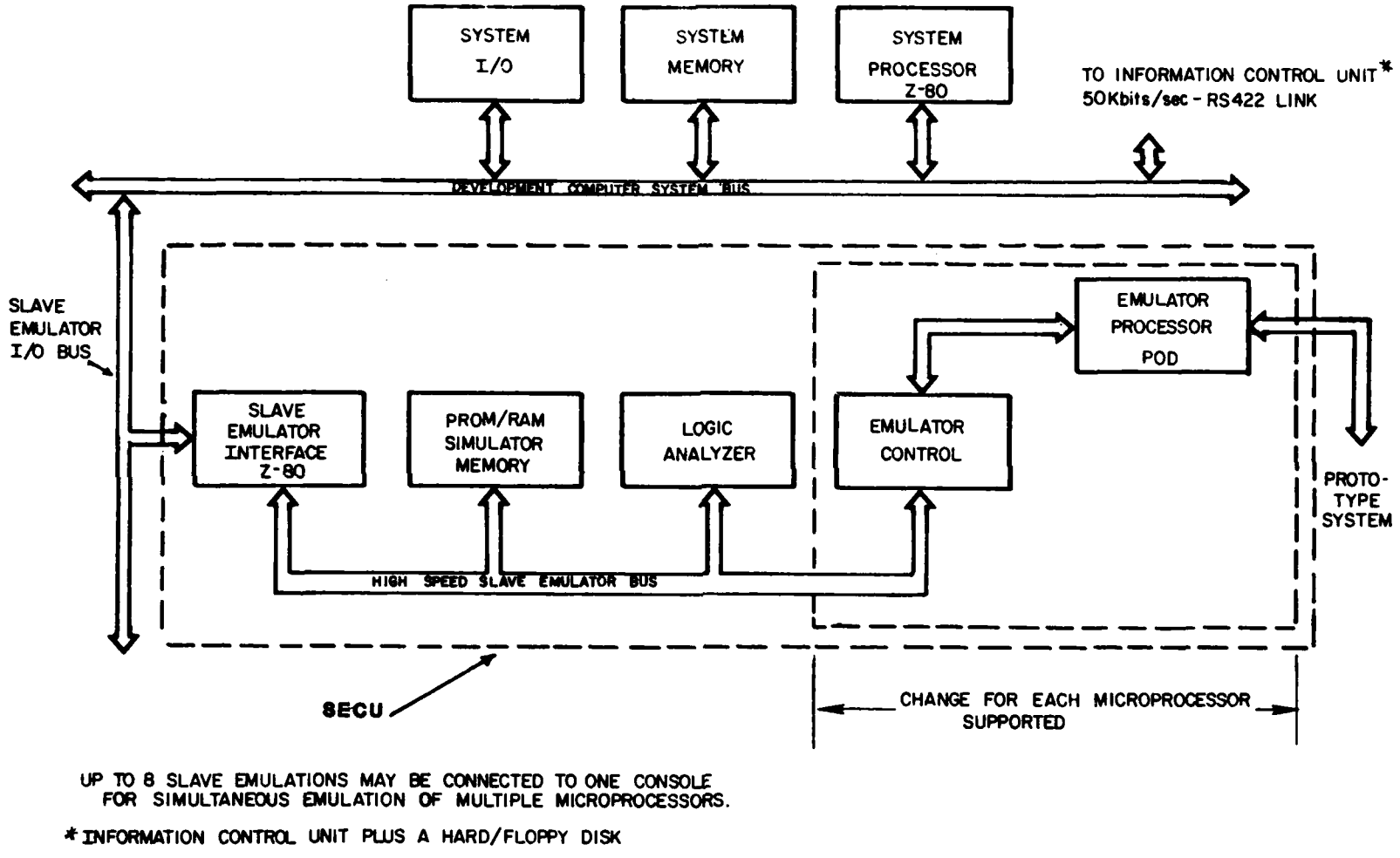
4.2.2 GenRad Advanced Development System Bus Structure

The GenRad multibus structure ensures real time operation of high-speed processors (figure 3) and is composed of the following buses:

Interface Bus - this bus, driven by the Z80 interface processor, communicates with the major modules of the SECU. In-circuit emulation controls down-line load/up-line save and memory mapping. Logic analyzer controls are handled over this bus.

Emulator Bus - high speed intermodule communication occurs over this bus between the target processor, emulator memory, logic analyzer and SECU control and status buffers.

I/O Bus and RS-232C Interfaces (19.2K bits/sec.) - tie the SECU to the GenRad 2300 series Advanced Development System console. Up to eight SECU units can be "daisy chained" to a single development system. Daisy chaining facilitates debugging of multiprocessor systems; that is, break events generated in one SECU domain can trigger debugging routines in other slave processors, allowing system-wide snapshots from a window which covers the multiprocessor



UP TO 8 SLAVE EMULATIONS MAY BE CONNECTED TO ONE CONSOLE FOR SIMULTANEOUS EMULATION OF MULTIPLE MICROPROCESSORS.

*INFORMATION CONTROL UNIT PLUS A HARD/FLOPPY DISK

Figure 3. GenRad Universal Advanced Development System Architecture

circuitry. A universal logic analyzer module (optional) is available for each SECU. The logic analyzer has three 48-bit breakpoints and has a trace RAM of 256 words x 48 bits.

4.2.3 GenRad Low Cost Development System

GenRad makes a less expensive system referred to as the Low Cost Development System (figure 4). This system is the entry level product to the complete GenRad family of development tools. It provides complete hardware and software development for 8-bit microprocessors such as Z80, 8080, 8085, 6800 etc. The low cost system consists of a console and floppy disk drives and utilizes a single processor architecture (figure 5). The system uses a single microprocessor to perform emulation up to 4 MHz with real time operation. During this test time, the processor handles only the prototype tasks, so no utilities can be serviced while emulation is in progress.

Priority interrupts will stop emulation allowing system calls to be serviced. Since only one processor handles all tasks over one bus, system operational speed is limited. This unit can be upgraded to the multiple bus architecture by adding a personality module in the system console and a SECU. All of the system capabilities of the advanced system are available when the upgrade is installed (reference 19).

4.2.4 FutureData

In February, 1982 FutureData introduced three new design tools (reference 20):

- o Co-emulation: A new concept which supports simultaneous emulation of closely coupled processor pairs. The system can provide emulation for the Intel 8086/8087pair, 16-bit microprocessor and Floating Point Processor (FPP), and the Intel 8088/8087, 8-bit microprocessor and FFP.

- o 8-MHz emulation support for the Intel 8086 microprocessor.

- o 5-MHz emulation for the 8088 and 8085 8-bit microprocessor.

In September 1982, the GenRad FutureData Division was sold to Kontron Electronics. Kontron expects to expand the FutureData Divison products. Kontron is a major designer of scientific instruments for the electronics development industry. This recent merger of product lines has resulted in a number of new design projects (reference 21).

4.2.5 GenRad/Futuredata Software Support

The Unified Disk Operating System (UDOS) (reference 22) will support a microcomputer system designed in either a single or multiuser environment. UDOS contains the following utilities:

Single/Multiuser File Manager - allows the user to specify file types, create, delete, move and display files on a CRT and/or a printer. The file manager provides data exchange and storage utilities to handle floppy diskettes and/or a Winchester disk.

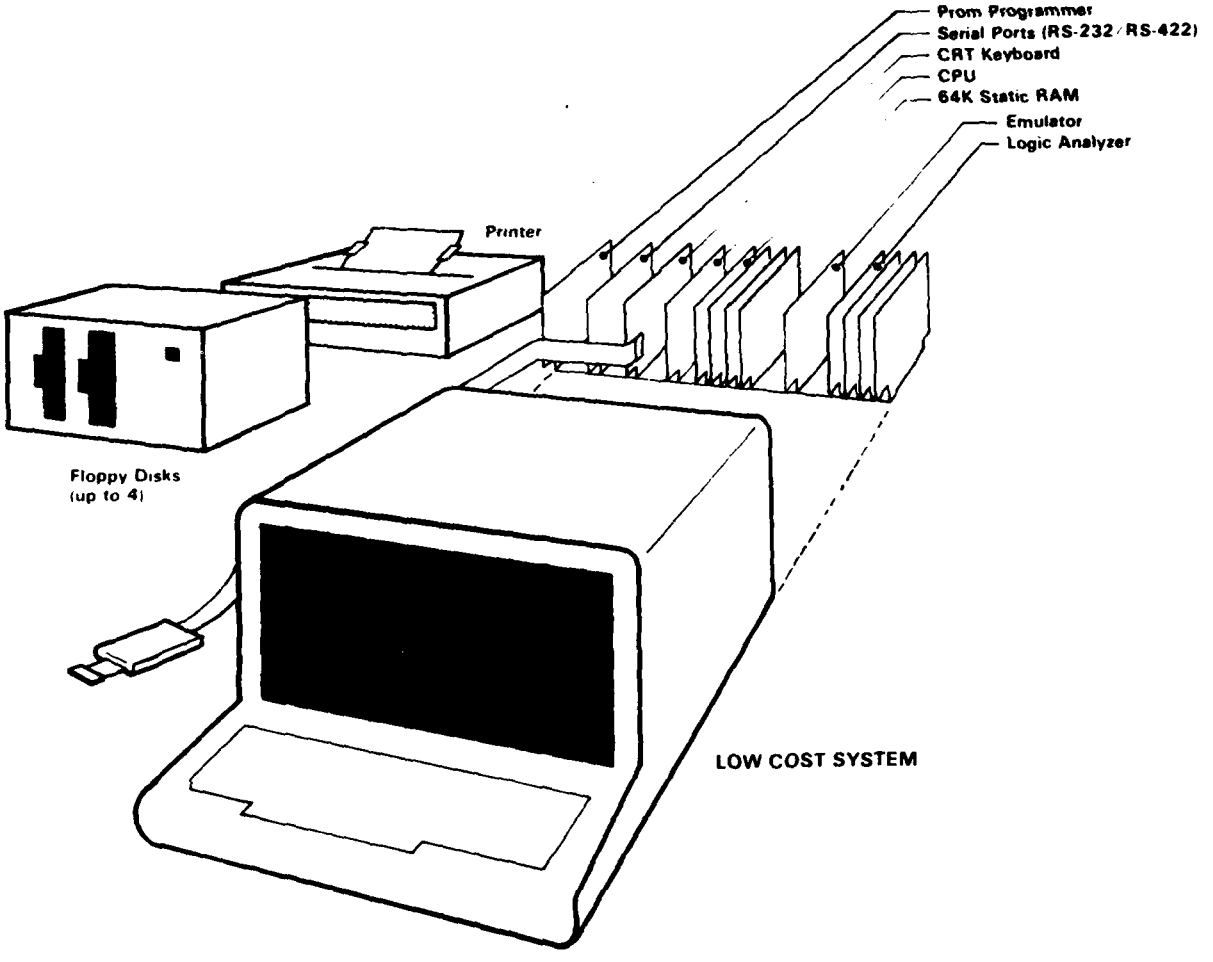


Figure 4. GenRad Low Cost Development System

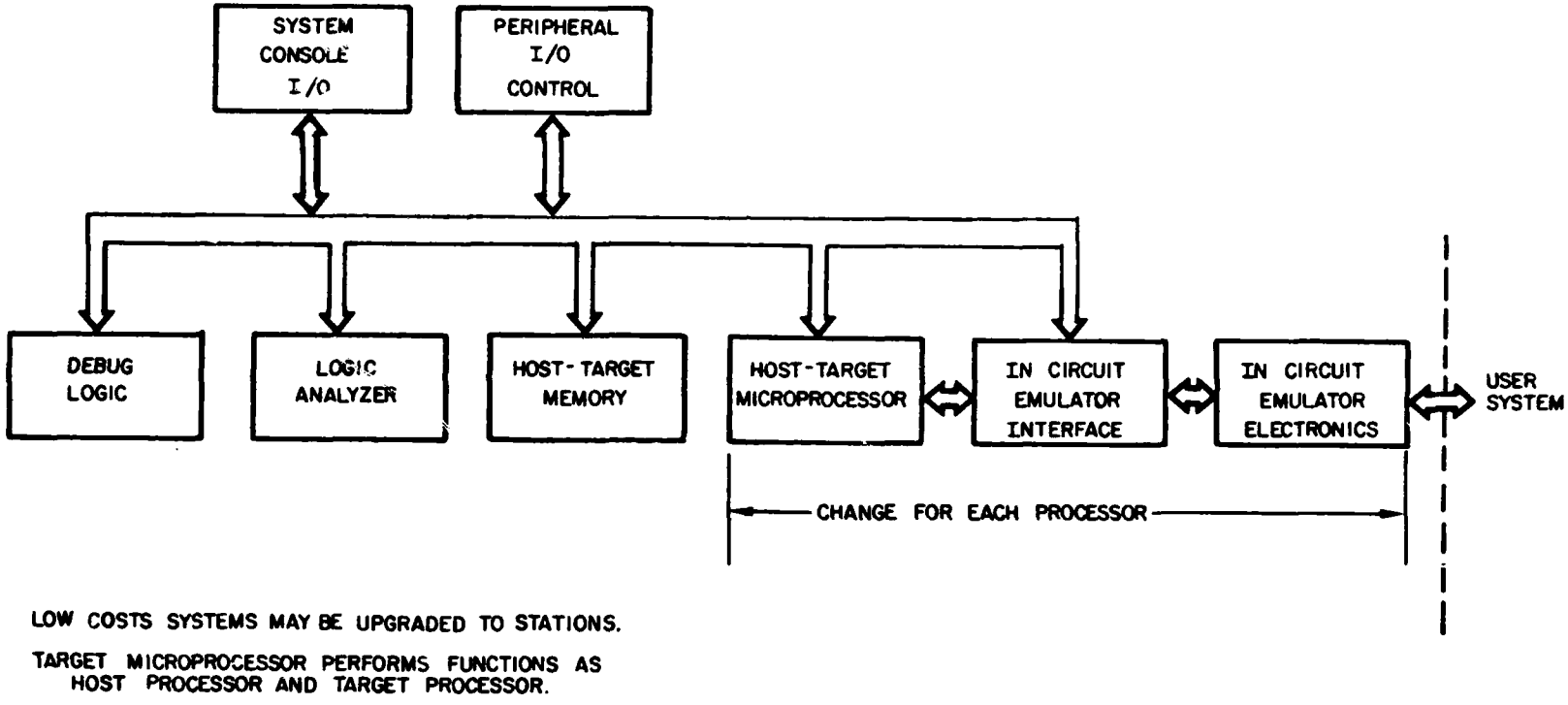


Figure 5. Single Processor Architecture

Text Editor - UDOS full screen editor aids the entry and changing of source files. Special function keys perform text scrolling, cursor positioning and character entry and deletion.

Relocatable Macro Assembler - a GenRad macro assembler is available for all of the microprocessors supported (see appendix II). Macro codes, conditional assembly, and psuedo-op constructs are common to all assemblers.

Console Debugger - the UDOS debugger features symbolic debugging, four software breakpoints and one hardware breakpoint.

Link Editor - the linker resolves intermodule references and can accept up to 255 relocatable program segments.

Command Control Language Processor - UDOS allows frequently used sequences of keyboard system commands to be stored and subsequently invoked via a single keystroke.

Intersystem Communications Processor - the UDOS software is capable of performing up-line saves and down-line loads to remote devices (computer, intelligent terminal, printer, etc.).

Binary/Intel Conversion Processor - this conversion utility can accept Intel binary object files from an Intel system and convert them to the GenRad/Futuredata format. Reverse conversion is also available.

4.3 HEWLETT-PACKARD

The Hewlett-Packard 64000 Series Development System in the standalone configuration, has been available only as a Winchester

hard disk-based system. A newly announced system, the HP 64110A has two floppy disk drives and an optional new logic state/software analyzer. The HP 64000 video terminal features "soft" keys which help the user quickly become familiar with the system. Reference to the HP system manuals is reduced to a minimum because of the menu-driven "soft" key features (reference 23). The HP CPU system control microprocessor is a HP proprietary device, not currently available for the OEM market.

Each HP development station is a complete unit which incoporates the 16-bit host processor, four buses, 128K of memory and a small tape cartridge drive. Up to six of these stations can be operated from a 20-120 megabyte disk drive. Also included with each station is a PROM programmer which can program up to 4K x 8 PROMs, and a logic analyzer. The HP 64005 workstation does not include a PROM programmer. The internal architecture allows the 16-bit host processor access to the system memory via a "mainframe" bus. The host processor can:

- o Configure system memory
- o Set up emulator run conditions
- o Set up the internal logic analyzer
- o Transfer data to/from I/O ports
- o Perform background tasks during real time emulator runs.

4.3.1 HP 64000 Bus Structure

The key to the basic station is the new HP 64000 multibus architecture (figure 6) which includes:

Mainframe Bus - carries most of the control, command and development station traffic from the control console CPU to/from the emulator.

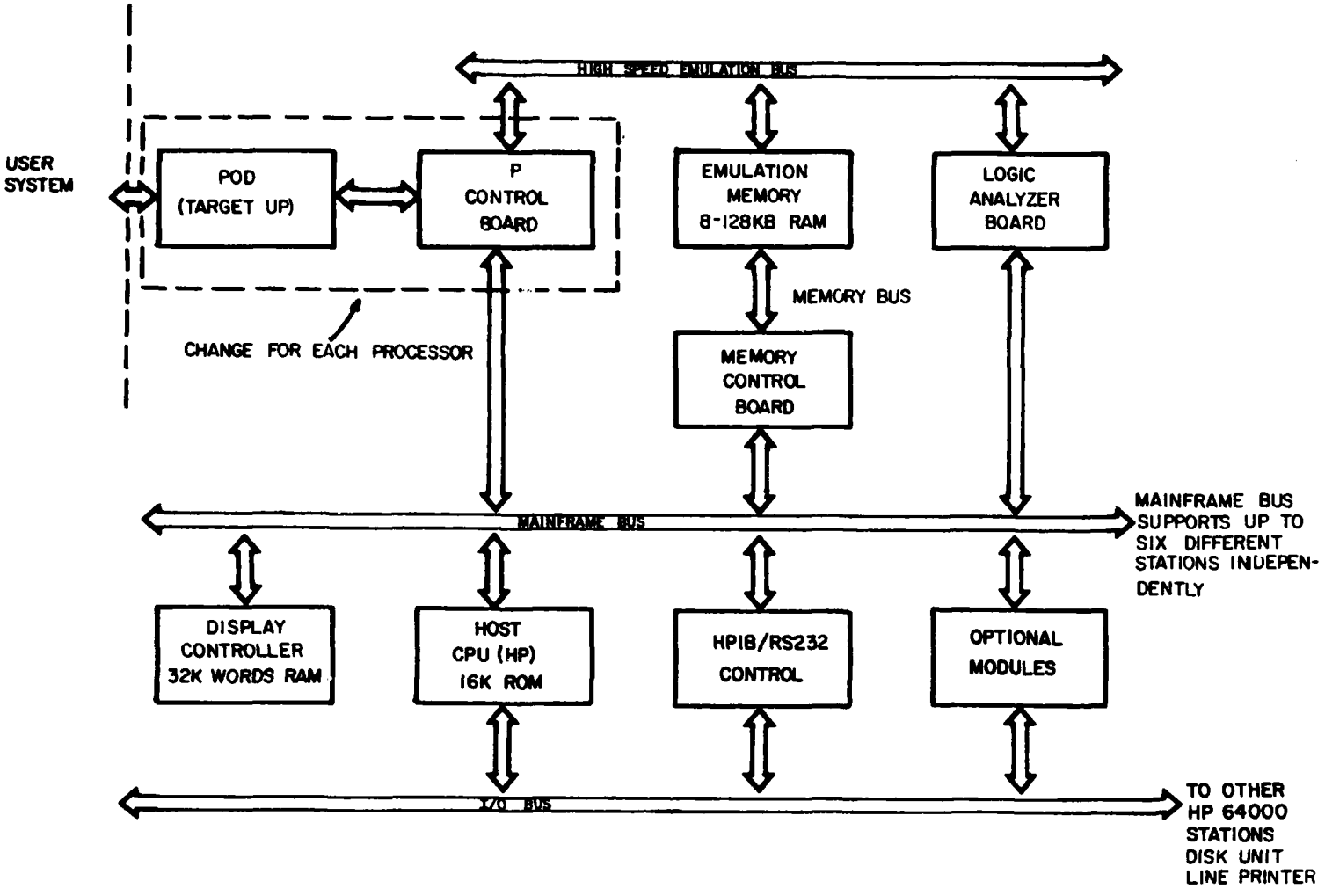


Figure 6. Hewlett-Packard Development System Architecture

I/O Bus - links host processor, keyboard, tape drive and I/O ports:

RS-232C port for general use,

Hewlett-Packard Interface Bus (HPIB) port is optimized for high speed data transfers to/from the Winchester disk.

Emulation/Memory Bus - ribbon cables connect the pod from emulator card to prototype system for in-circuit emulation. The HP 1611A logic analyzer board included in this system is dedicated to this high-speed bus.

Memory Bus - allows both the target microprocessor and host processor access to the emulation memory.

The HP system provides real-time emulation:

- o No imposed wait states
- o Bus activity identical to that of the prototype processor.

4.3.2 Software Support

Assembly language support is available for the following microprocessors:

- o Intel 8080/8085
- o Zilog Z80
- o Motorola 6800

PASCAL compilers are available for each of the microprocessors listed.

- o Intel 8080, 8085, 8086, 8088
- o Zilog Z80, Z8001, Z8002
- o Motorola 6800, 6809, 68000

HP Host PASCAL allows users to write PASCAL programs and utilities which execute on the development station's 16-bit host processor.

4.4 TEKTRONIX

Depending on the spectrum of capabilities and type of connectivity users require, they can select from the three types of Tektronix 8500 series Microprocessor Development Laboratories (MDLs) discussed here. The Tektronix 8540 interfaces with the users' mainframe, the 8550 is a standalone system, and the 8560 is a multiuser system.

4.4.1 8540 Integration Unit

The Tektronix 8540 integration unit interfaces with the users' host computer via ASCII RS-232C communications ports. This interface feature allows the user to take advantage of all of the mainframe multitasking facilities which may be already in-house. Real time emulation can proceed using the 8540 once the assembled/compiled object code has been down-line loaded from the user host.

4.4.2 8550 Standalone System

The Tektronix 8550 is a single-user system which provides all of the standard tools (assemblers, editors, linkers, loaders, and library files) to facilitate software development. The system's four microprocessors handle the development tasks:

- o LSI-11/23 - runs the Tektronix operating system DOS/50 which occupies 32K bytes of memory.

- o Z80A - dedicated to assembly, compilation, editor software handling, and a universal language translator used with appropriate cross-assembler.

- o Z80 - resides on the emulator board used for the particular target microprocessor being tested in the prototype circuit board. The emulator controller handles breakpoints and directs the operation of the emulator, thus allowing the emulator to be optimized for each target CPU. Support is available for both 8-bit and 16-bit chips.

- o 8085 - used for PROM programming.

4.4.3 8560 Multiuser Microprocessor Development Lab (MDL)

The Tektronix 8560 multiuser MDL is a hard disk-based system which will allow up to eight hardware or software workstations to provide multivendor emulator support. A UNIX-based operating system known as TNIX, derived directly from Bell Telephone Laboratories

UNIX V.7 operating system, runs the Tektronix 8560 multiuser software development unit (reference 24). TNIX includes:

- o Hierarchical filing system which allows files to be organized by project in sub-directories
- o Multilevel read/write protection
- o Text processing - all design specifications are easily updated to working documents
- o "Make" utility - tracks dependency between source and object modules, ensuring that only up-to-date object modules are included in the final program
- o Debug commands which are fully integrated into TNIX.

4.4.4 Tektronix 8500 Series Bus Structure

The Tektronix Microprocessor Laboratory (MDL) multiple bus architecture is illustrated in figure 7. The Backplane Bus is a medium-speed bus for handling system utilities. The Top Plane Bus is a high-speed bus that handles target microprocessor emulation, memory, and logic analyzer functions (reference 25).

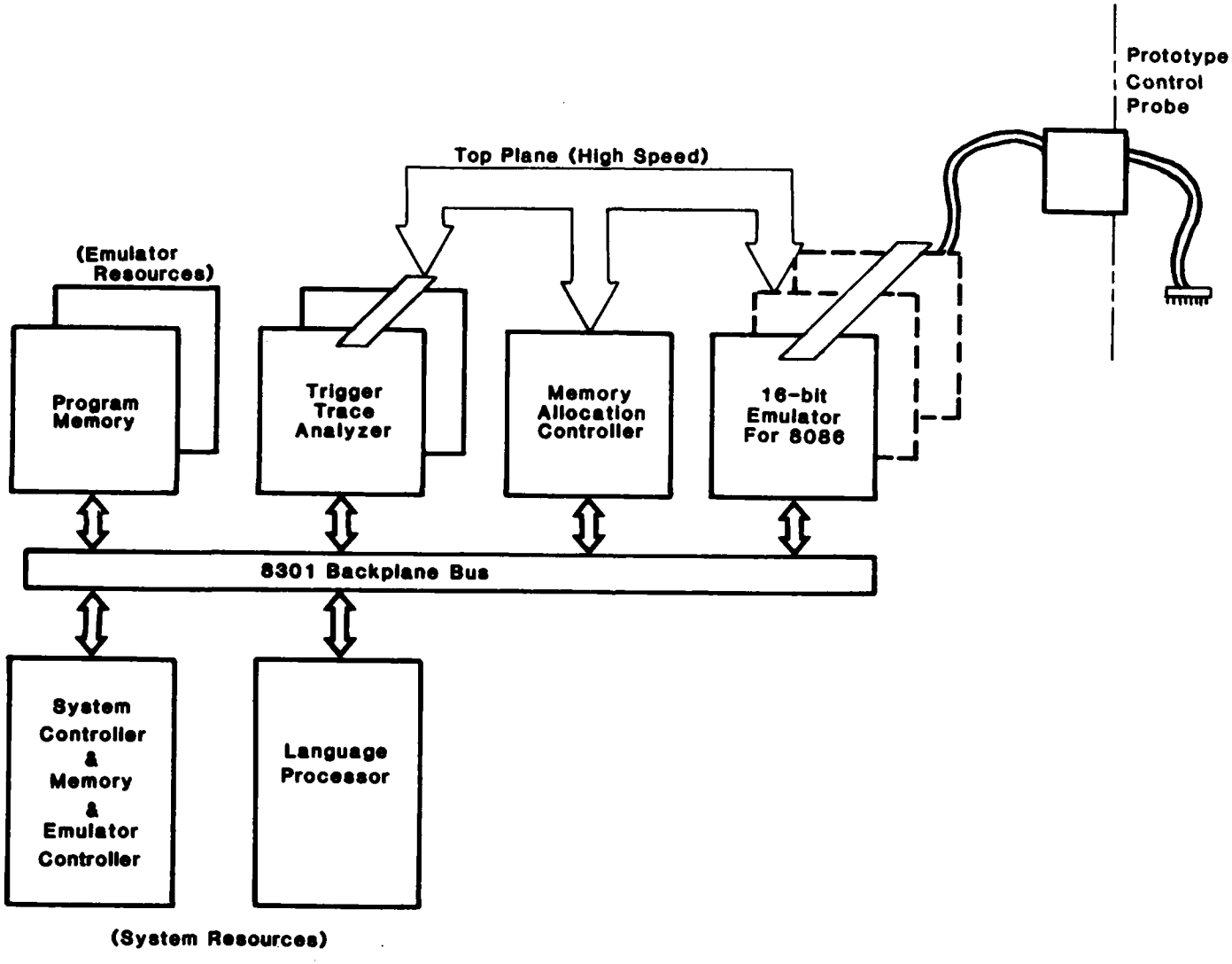


Figure 7. Tektronix Microprocessor Development Laboratory Architecture

SECTION 5

SUMMARY

The study of microprocessor development systems on today's market has produced the following conclusions:

- Microcircuit manufacturers produce development systems of their own logic design and dedicated to their own family of chips.

- Major instrument houses produce microprocessor development systems which can adapt to most of the microprocessors in general acceptance by industry today.

- When new microcircuit devices are being developed, the chip vendor, whose support system development has been going on concurrently with the new chip design, will probably reach the marketplace with a development system more quickly than will an instrument house.

- Designers who anticipate using more than one vendor's family of microprocessor chips and development systems should consider selecting a universal microprocessor development system.

- Three electronic measurement systems corporations who have produced microprocessor development systems which have been broadly accepted by the electronic industry are Gr[illegible]d, Hewlett-Packard, and Tektronix.

Each of these three manufacturers are producing microprocessor development systems which are based on a multiprocessor, multiple bus emulation system. Each of these systems can be obtained in an unbundled configuration, i.e., software can be developed on an existing mainframe, then downline loaded for in-circuit prototype system development. Each produces a system which should perform adequately when used to develop and debug a microprocessor-based system design.

- o The system attributes vary with each vendor's design. The prototype circuit under development should run at exactly the same speed whether or not a vendor's development system is running in-circuit emulation. If the selected development system cannot satisfy this requirement, an alternate development system which provides true real-time emulation should be seriously considered.

- o System costs vary even through development attributes are similar. See appendix III "Development System Costs."

- o As the user's experience with the development system and microprocessor grows, cross assemblers/compilers can be added to any mainframe. The multiuser capabilities of the mainframe can be used simultaneously to develop object code for various microprocessors. The object code can be loaded into the appropriate emulation station. This stepped progression of capability will provide an orderly transition from a single-user development system to a system which will allow multiple users to write software. Multiple emulation stations can be added as required.

- Microprocessor system complexity is growing rapidly. New microprocessor development systems are being announced at a similar rapid rate. Companies introducing new universal development systems and enhancements to the market are listed in appendix iv.

REFERENCES

1. Max Schindler, "Can One uC Development System be 200 Times More Powerful than Another? Shop and See." Electronic Design, 25 December 1978, pp. 60-67.

2. Martin J. Weisberg, "uP Development Systems," Digital Design, December 1979, pp. 81-84.

3. Girish Mhatre, "Microprocessor Development Tools," Electronic Engineering Times, February 15, 1982, pp. 39-58.

4. Girish Mhatre, "Microprocessors: The Issues," Electronic Engineering Times, September 15, 1980, pp. 88-104.

5. "PASCAL Package UPS Productivity," Electronics, October 20, 1982, p. 269.

6. Kurt Maly, Alan Hanson, "Fundamentals of the Computing Sciences," Prentice Hall, Inc., Englewood Cliffs, NJ, 1978, pp. 116, 117.

7. Jerry Miller, "68000-Based System Features CP/M Capabilities," Mini Micro Systems, October 1981, pp. 133-136.

8. Rebecca Thomas and Jean Yates, "A User Guide to the UNIX System," Osborne/McGraw-Hill, Berkeley, CA, 1982, pp. 1-13.

9. M. Elmor, D. Miller, J. Shwabe, "The Impact of 16-Bit Microprocessors on Development Software Tools," Computer Design, June 1981, pp. 111-115.

10. "Technical Summary, VAX 11/780," Digital Equipment Corporation, Maynard, MA, 1978, pp. g1-g16.

11. Kurt Maly, Alan Hanson, "Fundamentals of the Computing Sciences," Prentice Hall, Inc., Englewood Cliffs, NJ, 1978, pp. 267, 268.

12. "Logic Analyzers," Hewlett-Packard Measurement/Computation, Hewlett-Packard, 1982, pp. 160-193.

REFERENCES (Continued)

13. M. Mikalik and R. Francis, "Understanding Emulator Use to Increase Prototyping Skills," EDN, June 1981, pp. 121-126.

14. Bruce Gladstone, "Comparing Microcomputer Development System Capabilities," Computer Design, February 1979, pp. 83-90.

15. Bryan Call, "Emulators Can Help Speed Development," Computer Design, October 1982, pp. 79-82.

16. Keith Baker, "Microprocessor and Software Design Tools," Microprocessor and Microsystems, March 1979, Vol. 3, No. 2, pp. 87-92.

17. Edward R. Teja, "In-circuit Emulators Aid Designers as They Move From 8- to 16-bit Processors," EDN, August 1982, pp. 65-75.

18. "Multiuser Development System Meets Design-Team Needs," EDN, November 1981, p. 299.

19. GenRad Development System Product Summary, GenRad Data Sheet DS2025, March 1981.

20. "Extended Development System Tools Completely Support Intel uPs," EDN, February 17, 1982, p. 226.

21. "Kontron Electronics," Electronic Engineering Times, September 13, 1982, p. 165.

22. "FutureData Single/Multiuser Unified Disk Operating Systems," FutureData Data Sheet DS2029, January 1982.

23. T. Saponas, "Development System: Soft Keys + Four Buses = Easy Use and Full-Speed Emulation," Electronic Design, Vol. 20, 27 September 1979, pp. 38-42.

24. "TEK 8560 MultiUser Software Development Unit," Electronic Engineering Times, January 18, 1982, p. 71.

25. Ron Imbriale, "System Family Covers All uC-Development Needs," Electronic Design, October 11, 1980, pp. 141-145.

REFERENCES (Concluded)

26. Stan Baker, "UNIX-like OS Runs Tektronix' Development Tool," Electronic Engineering Times, November 9, 1981, pp. 12, 16.

27. "TEK Dedicates the 8500 Series," Electronic Engineering Times, February 15, 1982.

28. "Measurement Computation News," Hewlett-Packard, July/August 1982, p. 1.

29. A. Rappaport, "Integrated Engineering Workstations Simplify Complex uP System Design," EDN, March 13, 1982, pp. 41-46.

30. John Miklosz, "Test/Measurement Test Points," Electronic Engineering Times, October 11, 1982, p. 38.

31. Richard W. Comerford, "New Tools Add Design Power," Electronics, October 20, 1982, pp. 106, 107.

32. A. Santoni, "Special Report: Microprocessor Development Systems," EDN, April 28, 1982, pp. 86-97.

33. "Enhanced PASCAL Eliminates Need to Write in Assembly Language," Electronic Engineering Times, September 1982, p. 32.

APPENDIX I

MICROPROCESSOR DEVELOPMENT SYSTEM FEATURES

	GenRad	Hewlett-Packard	Tektronix
Real Time Emulator	Yes	See Note 1	Yes
Full Memory (64K) Address Range	Yes	Yes	Yes
Memory Mapping Over Full Address Range	Yes	Yes	Yes
Memory Write Protect	Yes	Yes	Yes
Simulated I/O Available in all Emulation Modes	No	No	Yes
Number of Emulation Breakpoints	4	15-20	6
Single Step Operation	Yes	Yes	Yes
Screen Editor	Yes	Yes	Yes
Change Microprocessor Support by Replacing Emulator Card and Probe	Yes	Yes	Yes
Directed Menu-Driven Control	Yes	Yes	Yes[2]
Floppy Disk Storage	Yes	Yes	Yes[3]
Cartridge Tape	No	Yes[4]	No
Downline Load/Upline Save of Data Files to Host System	Yes	Yes	Yes
Trigger Trace Analyzer (words x bits)	256 x 48	256 x 48	255 x 62

APPENDIX I (Concluded)

	GenRad	Hewlett-Packard	Tektronix
Number of External Analyzer Probes	8	8	8
PROM Programmer	Yes	Yes	Yes
Number of Multiuser System Stations With Hard Disk Support	8	6	8
Log File Capability	Yes	Yes	Yes
Disassembled File Display	Yes	Yes	Yes
High Level Languages:			
BASIC	Yes	Yes	Yes
PASCAL	Yes	Yes	Yes
Linking Loader	Yes	Yes	Yes
Macro Assembler	Yes	Yes	Yes

Notes:

1. No wait states when all memory assigned to target system (circuit under development). Possible wait states inserted when all program memory assigned to host system.

2. Tektronix has introduced "GUIDE," a user-friendly interface. GUIDE provides a menu-driven path through all system operations (references 26, 27).

3. The terminal contains a removable 225K byte cartridge tape for program storage. A new transportable microprocessor development station, HP64110A, with dual floppy drives, has been announced by HP (references 28, 29).

4. The HP 64005 can accept object code produced on a host system. The unit can then be used for complete emulation and in-circuit emulation. However, the HP 64005 cannot be used in a standalone configuration for program development.

APPENDIX II

MICROPROCESSORS SUPPORTED

GENRAD

16-Bit Microprocessors:

Intel 8086/8087
Motorola 68000
Zilog Z8000

8-Bit Microprocessors:

Intel 8080, 8085, 8088/8087
Mostek 3870
Motorola 6800, 6801, 6802, 6809
RCA 1802
Rockwell 6502
Zilog Z80

HEWLETT-PACKARD

16-Bit Microprocessors:

Intel 8086
Motorola 68000
Zilog Z8001, Z8002

8-Bit Microprocessors:

Intel 8080, 8048, 8049, 8085, 8088
Motorola 6800, 6802, 6809
Zilog Z80

TEKTRONIX

16-Bit Microprocessors - Assembler only:

Intel 8086
Motorola 68000
Texas Instruments TMS9900, SBP9900
Zilog Z8001, Z8002

8-Bit Microprocessors:

Fairchild F8
Intel 8080A, 8085A, 8088, 8049, 8048, 8035,
8039, 8039-A, 8021, 8041A, 8022
Mostek 3870, 3872, 3874, 3876
Motorola 6800, 6802, 6808
RCA 1802
Rockwell 6500/1

APPENDIX III

DEVELOPMENT SYSTEM COSTS

These costs were derived from the latest vendor specification sheets (April 1983) for a standalone system providing support for Z80 and 68000 design and development.

GenRad Universal Advanced Development System (Model 2301-9400)

Software development station with integral CRT display and keyboard

Z80 In-circuit Emulator and Z80 Probe

64K Bytes Static RAM

Logic Analyzer (48 bits x 256 words deep)

PROM Programmer for 2704, 2708, 2716, 2732 and 2532 EPROMS

Dual Floppy Disks: (Up to four single or double-sided double density drives can be supported by a standalone station).

Total Package Cost	$17,950

GenRad Optional Upgrade for 16-Bit System

Low Cost System to Console Slave Emulator Upgrade	3,550
Slave Emulator Chassis (SECU)	4,250
Includes: Separate chassis with eight card slots Power Supply RS-232 Cable Z80 Interface Processor	
2-32K Static RAMs ($1,850 each)	3,700
68000 Microprocessor Personality Probe and Assembler	4,450
Executable Trace Buffer Analyzer	1,850
Upgrade Total Cost	$17,800
8-Bit and 16-Bit Development System Total	$35,750

APPENDIX III (Continued)

Hewlett-Packard Development System (Model 64100)

Logic Development Station/Keyboard	$11,130
Dual Floppy Diskette	3,040
Z80 Real Time In-circuit Emulator	3,040
Z80 Emulator Software	100
Z80 Assembler	560
64K Bytes Emulator Memory System (200 ns access time)	3,540
Logic Analyzer (48 bits x 256 words deep)	2,530
PROM Programmer Subsystem	1,110
2732A PROM Module	300
Winchester 8" Disk (10 Mbytes)	4,360
Total Cost	$29,710

Hewlett-Packard Optional Upgrade for 16-bit System

68000 Emulator	4,300
68000 Software	100
68000 Assembler	1,060
Wide Memory Control Board	1,520
Upgrade Total Cost	6,980
8-bit and 16-bit Development System Total	$36,690

Note: Up to six development stations can share a common data base.

APPENDIX III (Concluded)

Tektronix Microcomputer Development Lab Model 8550

8550 Microcomputer Development Lab	$17,000
Z80A Assembler	1,000
Z80A Emulator Processor and Control Software	2,800
Z80A Prototype Central Probe	1,500
Advanced CRT-oriented Editor	500
Modular Development Language (MDLu)	1,900
64K Memory (55 nsec)	3,500
Trigger Trace Analyzer	4,500
CRT Terminal	2,700
PROM Controller	2,000
PROM Programmer Module (2716, 2732)	700
Total Cost	$38,100

Tektronix Optional Upgrade for 16-bit System

68000 Assembler	1,300
68000 Emulator Card	4,800
68000 Emulator Probe	2,700
Upgrade Total Cost	$ 8,800
8-bit and 16-bit Development System Total	$46,900

APPENDIX IV

LISTING OF VENDORS

Advanced Micro Devices, Inc.
901 Thompson Place
Sunnyvale, CA 94086 Telephone: (408) 732-2400

AM Sys 29/10 - Bit Slice Microcprocessor Development System which features an integrated logic analyzer. It provides a high-speed control-store emulator and target system clock control as well as logic state monitoring.

American Automation
14731 Franklin Avenue
Tustin, CA 92680 Telephone: (714) 731-1661

EZ-Pro Development System supports most well-known microprocessors including 8-bit and 16-bit micros plus bit slice microprocessors.

Applied Microsystems
5020 148th Avnue NE
Redmond, Washington 98052 Telephone: 1-800-426-3925

ES Series Satellite Emulator provides real time or single step emulation up to 6 MHz for different types of microprocessors. It also includes logic analyzer functions, disassembly, and memory mapping.

Boston Systems Office
469 Moody Street
Waltham, MA 02154Telephone: (617) 894-7800

Universal Microprocessor Development System (UMDS) consists of a number of software packages, assemblers, debuggers, linkers, etc. which are designed to run on DEC or Data General mainframes. All hardware must be obtained from other manufacturers.

Dolch Logic Instruments, Inc.
230 Devcon Drive
San Jose, CA 95112 Telephone: (408) 998-5730

The E80A Emulyzer traces microprocessor memory and internal register activities without the need of a full-fledged development system. A new Dolch product, Adaptive Test and Logic Analyzer System (ATLAS) is to be announced (reference 30).

Emulogic, Inc.
Three Technology Way
Norwood, MA 02062 Telephone: (617) 329-1031

ECL-3211 Microprocessor Development System. This system, built around a DEC PDP/11, can be made to emulate any chip of any word length, from any chip family. Hardware/software development capability is facilitated using this system.

John Fluke Manufacturing Co. Inc.
P.O. Box C9090, M/S 250C
Everett, WA 98026 Telephone: (206) 356-5400

The Fluke 9000 Series Microsystem Tester supports 32 different microprocessors.

GenRad/FutureData
Development Systems Division
5730 Buckingham Parkway
Culver City, CA 90230 Telephone: (231) 641-7200

See Kontron Electronics, Inc.

Gould, Inc., (Millenium Instruments Div.)
19050 Pruneridge Avenue
Cupertino, CA 95014 Telephone: (408) 996-9109

9520 Software Development System and 9508 Microsystem Emulator provide microprocessor hardware/software development tools.

Hewlett-Packard Company
1507 Page Mill Road
Palo Alto, CA 94304 Telephone: (213) 970-7500

HP 64000 Logic Development System. A complete integrated development system providing universal hardware/software development, emulation, analysis.

Hilevel Technology
14661 Myford Road
Tustin, CA 92680 Telephone: (714) 731-9477

Model DS270 provides a complete workstation for bit-slice processor designs.

Kontron Electronics, Inc.
5730 Buckingham Parkway
Culver City, CA 90230 Telephone: (231) 641-7200

A combination of the original company and GenRad's former FutureData Division have bundled the KLA 64 channel, 100 MHz logic analyzer, FutureData's 2302 emulator, a keyboard and floppy disk to form a programmable work station for hardware/software integration (reference 31).

Step Engineering
757 Pastoria Avenue
Sunnyvale, CA 94088 Telephone: (408) 733-7837

Step 4 - Real Time Memory Emulator allows user to develop software or solve hardware/software integration for both bit-slice and monolithic microprocessor systems (reference 32).

Tektronix, Inc.
Box 4828
Portland, OR 97208 Telephone: (800) 547-6711

Model 8560 Development System coordinates up to eight workstations sharing a 35 Mbyte Winchester disk. TNIX operation system supports text processing and documentation.

GLOSSARY

Breakpoint: A hardware or software condition (bit pattern) that stops program execution, e.g., specific addresses or control signals.

Cross-reference generator: A device that permits symbols (labels, variables, constants) to be correlated with their storage locations.

Cross software: Programs that permit a target system to be developed on a host computer with different CPU architecture.

Disassembly: Retranslation of machine language into mnemonics during debugging.

Down-line loading: Direct transfer of code from the host system into the target system or a PROM programmer.

Emulation: A hardware model of the target microprocessor is used by the development system to check out the target system. This can be either the same uP model as used in a target system or bit-slice architecture that mimics the target uP's function. Using the target uP is called substantial emulation, or in-circuit emulation.

Full-speed emulation: See "Real-time Emulation."

Hardware trace: Using a development system or logic analyzer in real time.

History: Preceding program steps, including address or register content, can be displayed when execution is stopped by an error or a breakpoint.

In-circuit emulation (ICE): See Emulation.

Interrupt handling: When the target uP's operation is stopped or resumed by the development system, no spurious signals are permissible. Maskable and nonmaskable interrupts must be handled as they would be in the target uC.

Linking loader: A routine that combines program segments and reassigns addresses to reflect the new memory locations.

Logic analyzer: Equipment that displays program timing and response signals. The trigger is normally a match with a specified bit pattern, or a signal that fills the logic analyzer's buffer. Individual probes that can be attached to any desired signal line greatly enhance the analyzer's power.

MDS: Microprocessor Development System.

Memory mapping: Replacing parts or all of the target system's memory with MDS memory without using special circuits in the target system.

Operating system: A general purpose high-speed data handling and file manipulation system which gives the user the ability to assemble, compile, link, relocate, execute and debug programs from a CRT terminal.

PROM: Programmable Read-Only Memory integrated circuit device.

Real-time emulation: The emulator works at the target system's speed - otherwise timing problems may be evident. This is especially important when the clock is linked with I/O decoding or used for noise reduction.

Relocatable macro assembler: A program that permits instruction groups to be combined for execution using symbolic addresses.

Signal matching: Inserting buffers near the target system when the target uP is replaced for emulation, so that the signals at the target uP are reproduced exactly.

Simulation: Modeling the target uP with a software interpreter so that object code can be checked as if it were actually executing in the target uP. Simulation usually cannot duplicate timing problems, glitches, or uP idiosyncrasies. I/O devices are often simulated so that uC development can proceed before the actual devices are available.

Soft keys: Unmarked keys on a keyboard which adopt different command functions at different points of the program entry process.

GLOSSARY (Concluded)

Software trace: Execution of one instruction at a time while the register status is displayed.

Suspended execution: Interruption of the target system's program, by the development system. However, signals may be needed to keep the target uP "alive." Chips like the Z80 can be suspended indefinitely by setting a control line.

Symbolic debugger: Refers to symbols and not addresses so the user does not need to refer to the memory map to locate the section of code he wants to debug. Symbolic debugger packages typically work with assembly language programs. A recent release of debugging software permits high level language such as PASCAL to be used for debugging code (reference 33).

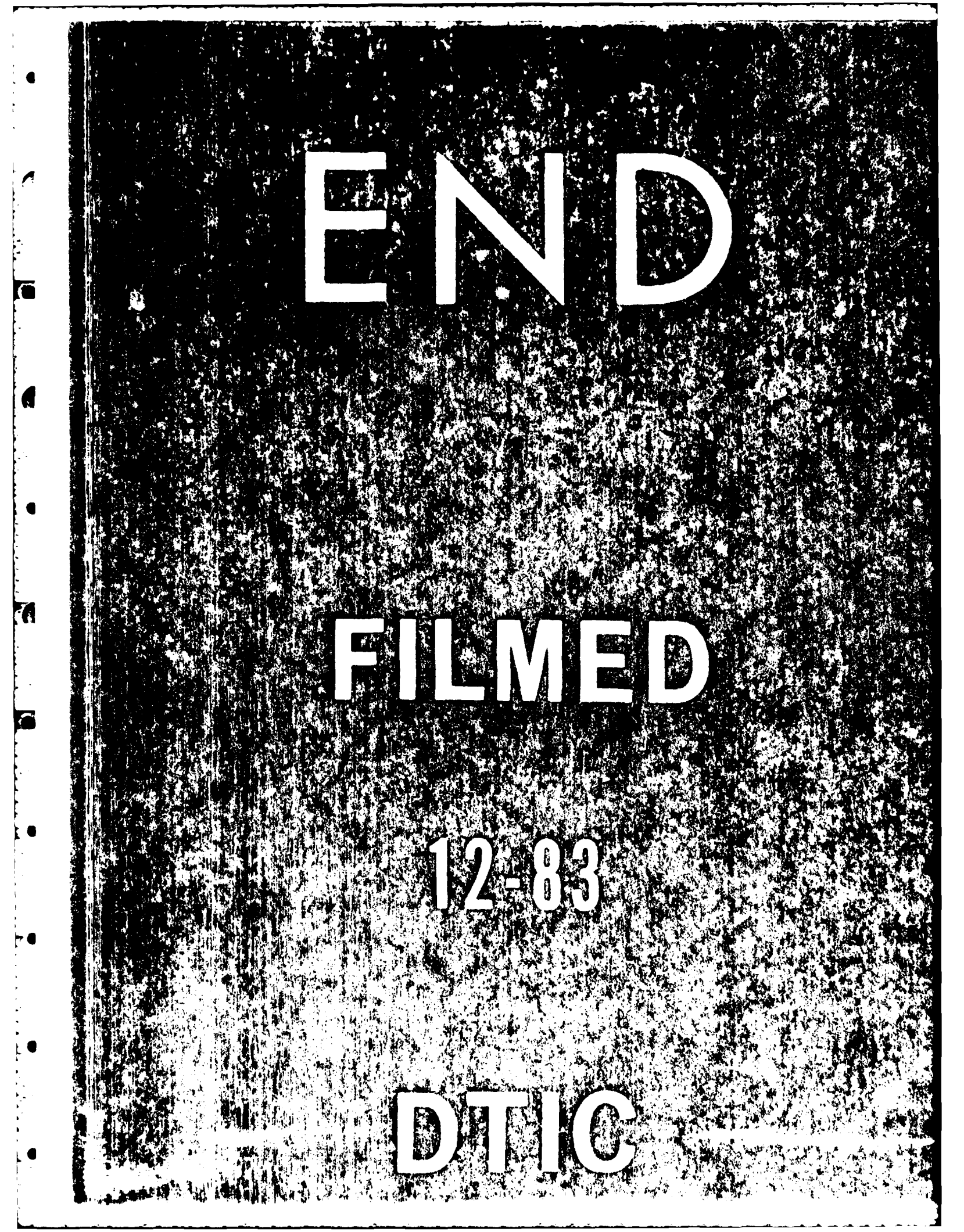
Target system: The microprocessor based system under development, the prototype.

Transparency: When the development system does not use or modify any of the target system's address or I/O space.

Universality: The ability of a development system to handle several uP families, either by exchanging CPU boards or by simulating a range of uPs.

Up-line save: Direct transfer of code from the target system into the host system. The host system can either be the development system or a mainframe system.

Verification: Checking out the target system's functions.


END
FILMED
12-83
DTIC